Claims

[c1]   1. A method for prefix encoding node identifiers in a logical tree comprising steps of:

a. choosing an initial base length with which to encode local identifiers,

b. assigning a value of zero as a node identifier to a root node in a logical tree,

c. sequentially assigning to descendants of a root node a local identifier having an even value and a length equal to said base length chosen in said choosing step, wherein said local identifiers are assigned in increasing value from leftmost children to rightmost children,

d. assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned, and

e. extending said initial base length if local identifier encoding combinations are exhausted before all descendants are assigned local identifiers.

[c2]   2. A method for prefix encoding node identifiers, as per claim 1, wherein inserting a node into an existing tree does not require change to existing node identifiers.

[c3] 3. A method for prefix encoding node identifiers, as per claim 1, wherein a node is inserted between a first node and a second node having consecutive local identifiers.

[c4] 4. A method for prefix encoding node identifiers, as per claim 3, wherein said inserted node is assigned a local identifier having a string length longer than string length of said first node.

[c5] 5. A method for prefix encoding node identifiers, as per claim 1, wherein assigning said node identifier to an inserted node comprises the following steps:

a. determining whether node to be inserted is inserted as a first child, between two existing siblings, or as a last child under a single parent node,

b. if said node to be inserted is inserted as a first child under said single parent node,

i. checking last byte of an existing first child,

ii. if the value of said last byte is not the smallest even number, then an even number greater than zero and less than the value of said last byte is selected to generate a local identifier of said node to be inserted, else

iii. if the value of said last byte of an existing first child is the smallest even number, generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number to generate a

local identifier and extending node identifier of said existing first child by a byte having a value of any arbitrary even number,

c. if said node to be inserted is inserted between two existing siblings under said single parent node, determining whether the string length of node identifier of said first sibling is less than, equal to, or greater than the string length of node identifier of said second sibling, else

d. if said node to be inserted is inserted as a last child after all other children under said single parent node, assigning to said node to be inserted an even local identifier greater than that of existing last child under said single parent node, and

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

[c6]   6. A method for generating and assigning prefix encoded node identifiers, as per claim 5, if said node to be inserted is inserted between two existing siblings under said single parent node and if the string length of local identifier of said first sibling is less than the string length of the local identifier of said second sibling,

a. checking if local identifier of said first sibling is the last available encoding value having a string length of

the local identifier of said first sibling and being smaller in value than said local identifier of said second sibling,

b. if said local identifier of said first sibling is the last combination having a string length of the local identifier of said first sibling that is smaller in value than said local identifier of said second sibling,

i. if the local identifier of said second sibling is not the first available identifier having the string length of the local identifier of said second sibling that is greater than the value of said local identifier of said first sibling; an even-valued local identifier being less in value than said local identifier of said second sibling and having string length of local identifier of said second sibling is generated and assigned, else

ii. generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number and extending local identifier of said existing first child by a byte having a value of any arbitrary even number less in value than said last byte of said existing first child, and

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

7. A method for generating and assigning prefix encoded node identifiers, as per claim 5, if said node to be

inserted is inserted between two existing siblings under said single parent node and if the string length of the local identifier of said first sibling is equal to the string length of the local identifier of said second sibling,

a. if the value of the local identifier of said first sibling plus two is less than the value of the local identifier of said second sibling, a local identifier for said node to be inserted takes on an even value greater than or equal to the value of said local identifier of first sibling plus two and less than the value of the local identifier of said second sibling,

b. if the string length of the local identifier of said first sibling plus two is equal to the string length of the local identifier of said second sibling, then the string length of the local identifier for said node to be inserted is extended wherein the length of the local identifier for the newly inserted node is the string length of said second sibling plus one, and the value of the first string length of said first sibling bytes is the node identifier of said first sibling plus one, and the new byte is an arbitrary even number less than the value of said last byte of the node identifier of said second sibling, and

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

[c8]   8. A method for generating and assigning prefix encoded node identifiers, as per claim 5,if said node to be inserted is inserted between two existing siblings under said single parent node and if the string length of the local identifier of said first sibling is greater than the string length of the local identifier of said second sibling

a. if the local identifier of said second sibling is not the smallest value having the string length of said second sibling that is greater in value than the local identifier of said first sibling, then a local identifier having a string length of said second sibling and having even value smaller than the value of the last byte of the node identifier of said second sibling is generated and assigned else,

b. if the local identifier of said first sibling is not the largest value with the string length of the local identifier of said first sibling, one of the larger values for the new encoding is generated and assigned, else

c. extending the local identifier of said first sibling by a length, by setting the last byte to the highest odd number and the new byte to an even number less than the value of the last byte, and

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

[c9]   9. An article of manufacture comprising a computer us-
able medium having computer readable program code
embodied therein which implements prefix encoding
node identifiers in a logical tree comprising modules im-
plementing code for:

a. choosing an initial base length with which to encode
local identifiers,

b. assigning a value of zero as a node identifier to a root
node in a logical tree,

c. sequentially assigning to descendants of a root node a
local identifier having an even value and a length equal
to said base length chosen in said choosing step,
wherein said local identifiers are assigned in increasing
value from leftmost children to rightmost children,

d. assigning to all subsequent nodes, node identifiers
generated by a concatenation of local identifiers of all
nodes along a path from a root node to a node to which
a node identifier is currently being assigned, and

e. extending said initial base length if local identifier en-
coding combinations are exhausted before all descen-
dants are assigned local identifiers.

[c10]   10. An article of manufacture comprising a computer us-
able medium having computer readable program code
embodied therein which implements prefix encoding
node identifiers, as per claim 9, wherein assigning a pre-

fix encoded node identifier to an inserted node comprises modules implementing code for:

a. determining whether node to be inserted is inserted as a first child, between two existing siblings, or as a last child under a single parent node,

b. if said node to be inserted is inserted as a first child under said single parent node,

i. checking last byte of an existing first child,

ii. if the value of said last byte is not the smallest even number, then an even number greater than zero and less than the value of said last byte is selected to generate a local identifier of said node to be inserted, else

iii. if the value of said last byte of an existing first child is the smallest even number, generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number to generate a local identifier and extending node identifier of said existing first child by a byte having a value of any arbitrary even number,

c. if said node to be inserted is inserted between two existing siblings under said single parent node, determining whether the string length of node identifier of said first sibling is less than, equal to, or greater than the string length of node identifier of said second sibling, else

d. if said node to be inserted is inserted as a last child

after all other children under said single parent node, assigning to said node to be inserted an even local identifier greater than that of existing last child under said single parent node, and

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

[c11]   11. An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements prefix encoding node identifiers, as per claim 10, wherein if said node to be inserted is inserted between two existing siblings under said single parent node and if the string length of local identifier of said first sibling is less than the string length of the local identifier of said second sibling, comprises modules implementing code for:

a. checking if local identifier of said first sibling is the last available encoding value having a string length of the local identifier of said first sibling and being smaller in value than said local identifier of said second sibling,

b. if said local identifier of said first sibling is the last combination having a string length of the local identifier of said first sibling that is smaller in value than said local identifier of said second sibling,

i. if the local identifier of said second sibling is not the

first available identifier having the string length of the local identifier of said second sibling that is greater than the value of said local identifier of said first sibling; an even-valued local identifier being less in value than said local identifier of said second sibling and having string length of local identifier of said second sibling is generated and assigned, else

ii. generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number and extending local identifier of said existing first child by a byte having a value of any arbitrary even number less in value than said last byte of said existing first child, and

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

[c12]   12. An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements prefix encoding node identifiers, as per claim 10, wherein if said node to be inserted is inserted between two existing siblings under said single parent node and if the string length of the local identifier of said first sibling is equal to the string length of the local identifier of said second sibling, comprises modules implementing code for:

a. if the value of the local identifier of said first sibling plus two is less than the value of the local identifier of said second sibling, a local identifier for said node to be inserted takes on an even value greater than or equal to the value of said local identifier of first sibling plus two and less than the value of the local identifier of said second sibling,

b. if the string length of the local identifier of said first sibling plus two is equal to the string length of the local identifier of said second sibling, then the string length of the local identifier for said node to be inserted is extended wherein the length of the local identifier for the newly inserted node is the string length of said second sibling plus one, and the value of the first string length of said first sibling bytes is the node identifier of said first sibling plus one, and the new byte is an arbitrary even number less than the value of said last byte of the node identifier of said second sibling, and generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

[c13]   13. An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements prefix encoding node identifiers, as per claim 10,wherein if said node to

be inserted is inserted between two existing siblings under said single parent node and if the string length of the local identifier of said first sibling is greater than the string length of the local identifier of said second sibling, comprises modules implementing code for:

a. if the local identifier of said second sibling is not the smallest value having the string length of said second sibling that is greater in value than the local identifier of said first sibling, then a local identifier having a string length of said second sibling and having even value smaller than the value of the last byte of the node identifier of said second sibling is generated and assigned else,

b. if the local identifier of said first sibling is not the largest value with the string length of the local identifier of said first sibling, one of the larger values for the new encoding is generated and assigned, else

c. extending the local identifier of said first sibling by a length, by setting the last byte to the highest odd number and the new byte to an even number less than the value of the last byte, and

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.